ERDC/GL TR-00-10

US Army Corps of Engineers®
Engineer Research and Development Center

Eagle Terrain Preprocessor

George B. McKinley, Terril C. Falls, and David C. Stuart

September 2000

Geotechnical Laboratory

Approved for public release; distribution is unlimited.

20001023 026

ERDC/GL TR-00-10
September 2000

Eagle Terrain Preprocessor

by George B. McKinley, Terril C. Falls

Geotechnical Laboratory
U.S. Army Engineer Research and Development Center
3909 Halls Ferry Road
Vicksburg, MS 39180-6199

David C. Stuart

Applied Research Associates
Vicksburg, MS 39180

Final report

Prepared for U.S. Army Corps of Engineers
Washington, DC 20314-1000

Under Project AMIP-97-WARF-0

Engineer Research and Development Center Cataloging-in-Publication Data

McKinley, George B.
Eagle terrain preprocessor / by George B. McKinley, Terril C. Falls, David C. Stuart ; prepared for U.S. Army Corps of Engineers.
34 p. : ill. ; 28 cm. -- (ERDC/GL ; TR-00-10)
Includes bibliographic references.
1. Eagle (Computer program) 2. Military topography -- Data processing. 3. Algorithms. I. Falls, Terril C. II. Stuart, David C. III. United States. Army. Corps of Engineers. IV. Engineer Research and Development Center (U.S.) V. Geotechnical Laboratory (U.S.) VI. Title. VII. Series: ERDC/GL TR ; 00-10.
TA7 E8 no.ERDC/GL TR-00-10

Contents

List of Figures

Preface

The study reported herein was conducted by personnel of the U.S. Army Engineer Research and Development Center (ERDC), Geotechnical Laboratory (GL), Mobility Systems Division (MSD), Vicksburg, MS. The work was conducted under the Project AMIP-97-WARF-01, "Application of the Terrain Evaluation Model (TEM) to Strategic-Level Campaign Analysis." Sponsors for the project were the U.S. Army Center for Army Analysis (CAA), Ft. Belvoir, VA, and the Training and Doctrine Analysis Command, Leavenworth, KS. The work was conducted between December 1997 and March 1998.

The study was conducted under the general supervision of Dr. Michael J. O'Connor, Director, GL, and Dr. William E. Willoughby, Acting Chief, MSD; and under the direct supervision of Mr. Walton C. Dickson, Acting Chief, Modeling and Simulations Branch (MSB), MSD. The logic and computer programming was accomplished by Messrs. George B. McKinley and Terril C. Falls, MSB, and David C. Stuart, Applied Research Associates, Vicksburg, MS. MSD personnel supporting this study were Messrs. Burhman Q. Gates and Conrad P. Rabalais and Ms. Mary A. Dungan. Messrs. McKinley, Falls, and Stuart prepared the report.

At the time of publication of this report, Dr. James R. Houston was Director of ERDC, and COL James S. Weller, EN, was Commander.

1 Introduction

Background

The US Army Engineer Research and Development Center (ERDC), Waterways Experiment Station, previously developed software (McKinley et al. 1993) which creates a network of mobility corridors (MCs). The edges comprising the network are given an attribute corresponding to the maximum friendly military unit that the edge can accommodate. This mobility corridor generation algorithm was selected as one of the original ten Tactical Decision Aids (TDAs) running on the Terrain Evaluation Module (TEM). TEM is an element of the Army Common Software Program and the Common Operating Environment and is already operational at the Concepts Analysis Agency (CAA). CAA presently uses several campaign simulation models, including the corps-level Eagle model, to carry out the multiple levels of war-fighting analysis required to complete studies and analysis. TRADOC Analysis Center and Los Alamos National Laboratory developed the Eagle combat simulation model. The terrain and maneuver environment in Eagle is based on a network representing tactical MCs for companies and battalions. This network is similar to that created by the MC network generation TDA in TEM. The edges comprising the network required to run Eagle have several attributes, which were not computed in TEM. Eagle also requires terrain data defined with respect to a grid structure, usually 4 by 4 km, overlaying the battle area. The Eagle Graphics Interface Package requires a graphics image map for use as a background map during visualization of the battle. These terrain-related input data are currently generated by means of cumbersome and time-consuming methods. To develop a database for an Eagle application, approximately 36 to 50 continuous hours of computational time are required on a Sun Ultra system.

Purpose

The purpose of this report is to describe the automated user-friendly programs produced by the ERDC which create the terrain-related inputs required by the Eagle model. The project was divided into three tasks. The first task involved modification of the MC network creation software in TEM to compute the additional attributes required by the Eagle model and also to output the network in the format required by Eagle. Also, three methodologies are examined to discover

the approach that will create the most complete MC network. The second task was to produce software to create the Eagle terrain tiles and output them in the proper format. Task three required the creation of software to generate a graphics image in X-Windows dump format for use as a background image by Eagle. These programs are described in terms of their input, methodology, and output.

2 Mobility Corridor Network Generation

Classical Thinning Algorithm

The network of potential MCs is generated from a map matrix of predicted vehicle speeds. Three different thinning algorithms were tested for suitability to this application. The first tested was the classical thinning algorithm. This thinning algorithm thins the areas of the map matrix which have predicted speeds greater than a designated threshold. The pixels comprising those areas with speeds greater than the threshold are set to one and all other pixels are set to zero. A buffer one-pixel-wide is created around the edge of the map and is also set to zero. The classical thinning algorithm (Pavlidis 1982) was implemented as follows:

```
Set a pass counter N to 0
Set the flag REMAIN to true
While REMAIN is true do
    Increment N
    Set REMAIN to false {No change has been made}
    For J=0, 2, 4, and 6 do
        For all pixels P in the map matrix do
            If P is 1 and if its J-neighbor (Figure 1) is 0 then
                Set flag SKEL to false
                For all six patterns PA shown in Figure 2 do
                    If the neighborhood of P matches any of the patterns PA, then
                     {For a group of pixels to match a pattern, then at least one of each
                     group of pixels marked with A or B must be non-zero}
                        Set SKEL to true
                    End If
                End Do {For}
                If SKEL is true then
                    Set pixel P to 2 {skeletal pixel}
                    Set the pixel corresponding to P in another matrix to N
                Else
                    Set pixel P to 3 {deletable pixel}
                    Set REMAIN to true
                End If
            End If
        End Do {For}
        For all pixels P in the map matrix do
            If P is 3 then
                Set P to 0
```

```
            End If
         End Do {For}
      End Do {For}
   End Do {While}
```

3	2	1
4		0
5	6	7

Figure 1. Enumeration of neighbor pixels in the thinning algorithms

A	A	A
0	P	0
B	B	B

A	0	B
A	P	B
A	0	B

A	A	A
A	P	0
A	0	2

A	0	2
A	P	0
A	A	A

A	A	A
0	P	A
2	0	A

2	0	A
0	P	A
A	A	A

Figure 2. Patterns used in the classical thinning algorithm

A simple test case map was created to test the thinning algorithms. This map is shown in Figure 3. The output produced by the classical thinning algorithm is shown in Figure 4. Obviously there are many disconnects in this network and this algorithm seems unsuited to this application.

Patch Expansion

The first step in thinning by patch expansion is to transform the matrix of GO/NOGO speeds into a map of contiguous patches by use of a technique referred to as factoring. The factoring method used in this algorithm is a two-pass process. In the first pass, if two adjacent cells are both NOGO then they are assigned the same patch number. If two adjacent cells are found to both be NOGO but have different patch numbers, then those two numbers are flagged in an equivalence table. In the second pass through the map, all the cells of equivalent patches are set to the same patch number. The patches are numbered sequentially beginning with two. A buffer-one-pixel wide is created around the edge of the map and is set to one. The result of factoring a portion of the test case is shown in Figure 5.

After transforming the GO/NOGO matrix to a map of patches, the NOGO patches and the border are grown by means of cellular automata. In each step, every cell that is not part of a NOGO/border patch but is adjacent to a NOGO/border cell from a previous step, is made part of the NOGO/border patch. This process is repeated until all the GO areas are filled. The edges where the expanded NOGO/border patches meet constitute the MCs. The output produced by patch expansion on the test case is shown in Figure 6. Obviously several edges are left out of this network. This method leaves out all potential MCs that lie within a patch's perimeter. No edges result, since the same patch expands into the vacant pixels from different sides. The two large patches in the test case that meander around the page best exemplify this.

Modified Classical Thinning Algorithm

In an effort to develop an algorithm that will produce a network with few or no disconnects and also has edges that may fall within a patch's perimeter, the classical thinning algorithm was modified. This modification consisted of allowing the last four patterns in Figure 2 to be matched by either a value of one or two when previously there had to be two. The output produced by modified classical thinning on the test case is shown in Figure 7. Obviously there are no disconnects in the resulting network and the problems inherent in patch expansion are eliminated. Thus, this algorithm seems best suited to the automated generation of MC networks.

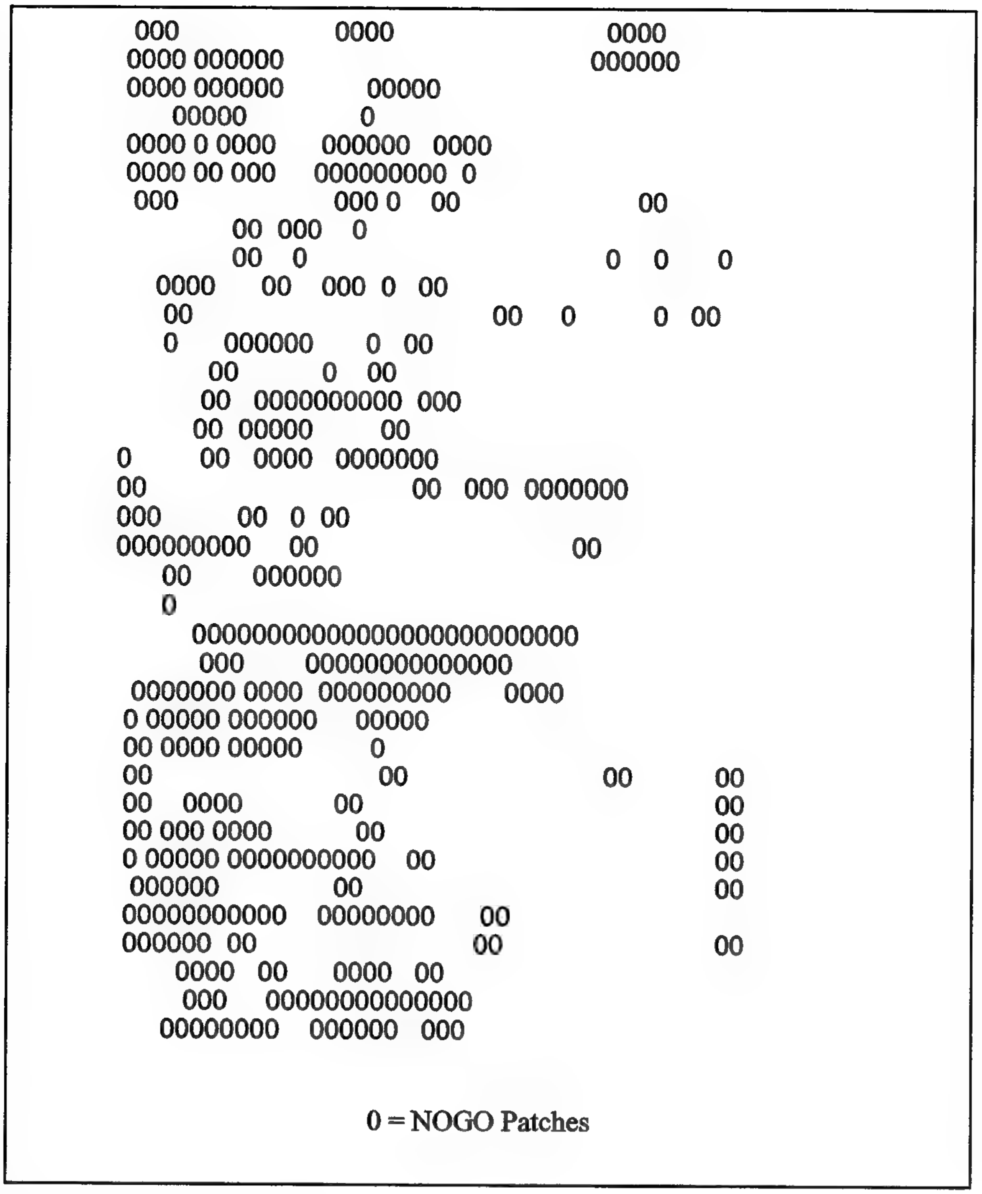

Figure 3. Test case map

Network Creation

The thinned areas are converted to an edge/node network format. The first step in this process is to find edges and nodes by summing the number of neighboring pixels that are non-zero for each pixel that is non-zero. Edges are the pixels that are assigned a sum of two and nodes are all other non-zero pixels. When summing the non-zero neighboring pixels, if a diagonal neighbor pixel is non-zero and either the vertical or horizontal neighbor pixel adjacent to that diagonal neighbor is non-zero then the diagonal neighbor pixel is not included in the sum. For example, if neighbor pixel 3 in Figure 1 were non-zero and either neighbor 2 or neighbor 4 were non-zero, then neighbor pixel 3 will not be used in the sum.

```
----------------------------------------
--00000-0000000-------000000000---------
-0--  0   ---0-----0---   --0--------
-0-   0     --000000--     --0000----
-0-   0     --0-----0--     ------0---
--00000-0     --0------000--- ------0---
-0-   0                  0   --0---     00-    ---0---
-0-   0                    0  -0---        0- -----0---
-0--  0                      0 --0-  -0000---  0--0000----
-0-----0---0--0--  0-   00-- -00----0---
-0---00-000-0-0--  0--- - 0- -0- ---0---
--0-0--    --0---  -0--   0- -0-  --0---
---0----  --0-----  -0--- 0- -0-  --0---
---0---- --0---      -00--0- ---  --0---
---00-----0---  -00000--00- ----  --0---
--0--0---0---  00-         0-   ---0---
-0----0-0---  0-     --00000--  ----0---
0- ----0-0--  0--    -0-       -----0---
0-  ------0--  00-   0-       ------0---
0-   ------0---  0-- 0-  ----------0----
0-         -0---  000--  -------000-0---
0-------  --000---      --000000-----0--
-0------ --0---00000000000------------0-
--00000---0-                         0-
-0-----0000--   -00000--              0-
-0--       0    --          -------   0-
-0- 0                     0      -----     ---000-----0-
-0-  0                    0     --------- ---0---00--0--
-0-  0--00-  ------0-----  --0--  -00---
-0-  -00-    ---000-00---  --0--  --0---
-0-  0                  0     --------0--  --0--  --0---
-0- 0                     0        -0--  --0--  --0---
--00      -000---00000---  --0--  --0---
-0-            0-0-        --0---  --0---
-0-       --  --0-  -----0000----  --0---
-0--------      ---  --000-     ---  --0---
-0----00---   -----              ---0---
--0000--         ----       ---   ----0---
------0----------00--------0-----000----
-------0000000000--00000000-00000-------
```

- Deleted Grid Cell

0 Retained Grid Cell

Figure 4. Result of classical thinning on test case.

```
    222 3333        4444
    2222 333333       444444
    2222 333333        44444
       33333          4
    5555 3 3333     666666  4444
    5555 33 333    666666666  4
     555 333 3   66                           66
            66 777    6
            66   7                        7 6       6
      8888      66    777  6   66
       88      66    7  6   66
       8     666666      6   66
          66          6    66
          66   6666666666  666
         66  66666        66
   9      66   6666   6666666
   99       66   666  6666666
   999        66   6  66
   999999999    66                        66
       99      666666
        9
```

Figure 5. Result of factoring on portion of test case

The next step toward network creation involves traversing the pixels that form each edge. Once the pixels forming an edge are identified, the nodes that are located at each end of the edge are stored. Then the largest formation that the edge can support is computed based on the average of the widest 90 percent of the pixels comprising the edge. The width associated with a pixel is based on the number of passes required to thin the area surrounding that pixel. It is assumed that the average number of thinning passes for the edge multiplied by the minimum of the X and Y cell resolutions is one-half the width of the MC represented by the edge. This assumption appears valid since the lines are generally reduced from two sides on each thinning pass. The widths used to class the edges are shown in Table 1. Edges having average widths less than 500m are assigned a company level, but these edges are also treated as a choke point for their entire length. Choke points are assigned to portions of other edges that are less than one-half the designated width of the unit that the edge can support. The final stage of network creation is accomplished by the conversion of the raster edges to vectors.

```
-------------------------------------------
---00000-0000000------0000000000--------
--0-   -0   ----0----0---    ---0-------
--0    -0     ---0000-0-     ---000----
--0                  0     --0------0000   ------0---
---0000-0     --0----------00 ------0---
--0    -0-    -0----       ---0   --0----
--0    -0 -   -0--- 00000   --0-000-0---
--0-   0   - -0--  0-----00  --0----0---
--0--00-00000-0--  0-    ---0 -------0---
--0-0--------0---  0--- - -0 --- ---0---
---0---    --0---  -0--   -0 ---  --0---
----0---  --0-----  -0--- -0 ---  --0---
----00-- --0---      -00--0- ---  --0---
---0--0---0---  --------00- ----  --0---
--0----0--0--  ---          --   ---0---
-0------00--  --     ---------  ----0---
-0 -------0-  ---    ---        -----0---
-0  -------0-  ---   --       -----0----
-0   -------0--  --- --  -------000-0---
-0        -000-  -----  -000000-----0--
-0------  -0---00000000000------------0-
--00000- -0---------------------------0-
--0----000--                      0-
--0----------   --------          0-
--0-        -    --         -------    0-
--0 -     -      -----     -----------0-
--0  -     -     ---------- -----------0--
--0  ------  ------------  -----  --0---
--0  ----    ------------  -----  --0---
--0  -   -    -----------  -----  --0---
--0 -     -          ----  -----  --0---
--0-      ----------------  -----  --0---
--0           ----        ------  --0---
--0       --  ----  -------------  --0---
--0-------    ---  ------     ---  --0---
---000-----   -----              ---0---
------0-        -00-     -0-   -000----
-------0000000000--00000000-00000-------
----------------------------------------

       -  Deleted Grid Cell
                      0                  Retained Grid Cell
```

Figure 6. Result of patch expansion on test case.

```
----------------------------------------
--00000-0000000-------000000000---------
-0--   0    ---0-----0---    --0--------
-0-    0     --000000--      --0000----
-0-    0     --0-----0--     ------0---
--00000-0     --0------000--- ------0---
-0-    0                 0    --00--     00-    ---0---
-0-    0                  0   -0--0                      0- -----0---
-0--   0                   0 --0-  00000---  0--0000----
-0-----0---0--0--  0-   00-- -00----0---
-0---00-000-0-0--  0--- - 0- -0- ---0---
--0-0--    --0---  -0--   0- -0-  --0---
---0----  --0-----  -0--- 0- -0-  --0---
---0---- --0---      -00--00 -0-  --0---
---00-----0---  -00000--00- 00--  --0---
--0--0---0---  00-         0-   ---0---
-0----0-0---  0-     --00000--  ----0---
0- ----0-0--  0--    -0-       -----0---
0-  ------0--  00-   0-      ------0---
0-   ------0--0                       0-- 0-  ----------0----
0-         -00--  000--  -------000-0---
0-------  --0-0--0      --000000-----0--
-0------ --0---00-00000000------------0-
--00000---0-                         0-
--0----0000--   -00000--               0-
-0-0       0                        0-          -------    0-
-0- 0                       0         0----      ---000-----0-
-0-  0                      0     --0------ ---0---00--0--
-0-  0---0-  ------0-----  --0--  -00---
-0-  -000     ---000-00---  --0--  --0---
-0-  0                    0     --------0--  --0--  --0---
-0- 0                       0         -0--  --0--  --0---
--00        -000---00000---  --0--  --0---
-0-            0-0-        --0---  --0---
-0-        --  --0-  -----0000----  --0---
-0---------     -0-  00000-     ---  --0---
-0----00---   -0--0             ---0---
--0000--        00--      ---   ----0---
------0----------00--------0-----000----
-------0000000000--00000000-00000-------
```

- Deleted Grid Cell
0 Retained Grid Cell

Figure 7. Result of modified classical thinning on test case.

Table 1 Widths Used to Class Edges	
Unit	**Width, m**
Corp	12,000
Division	6,000
Brigade	3,000
Battalion	1,500
Company	500

Once the network is created, dead-end edges less than 500 m long are deleted. Then edges that can support a unit larger than any of the other edges connected at either of that edge's nodes are given a lower unit assignment. For example, if an edge may support a battalion, but all other edges connected at each of that edge's nodes only support a company, then that edge will also be set to a company. These last two steps are included to produce a simpler network. An example of a MC network for the National Training Center (NTC) is shown in Figure 8. For this example, urban areas and speeds less than or equal to 1 kph were set to zero.

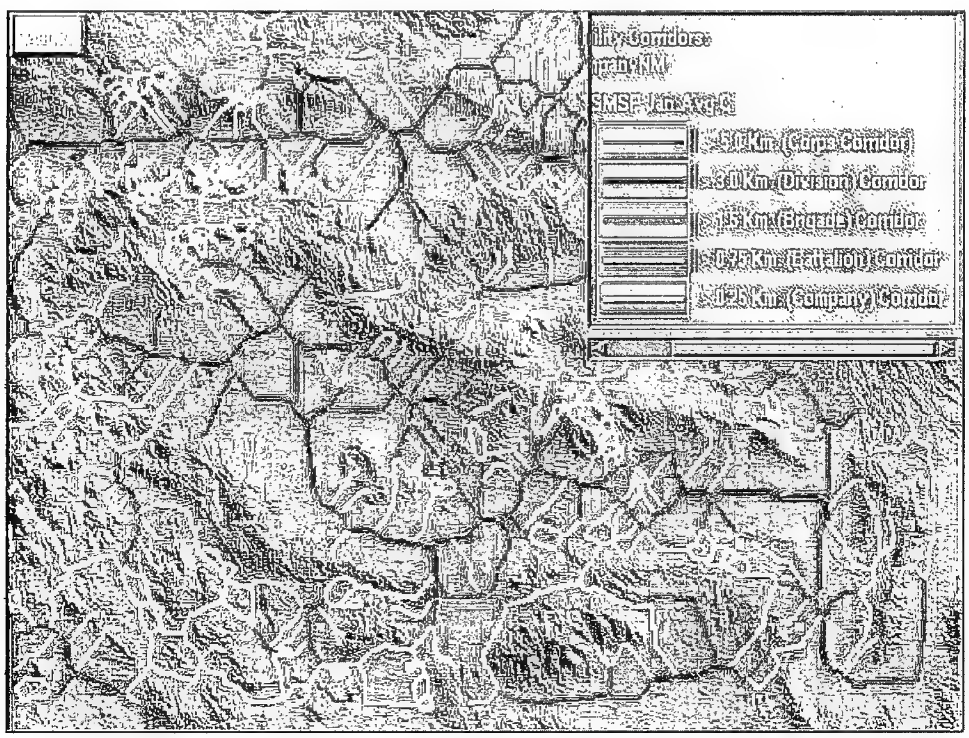

Figure 8. Mobility corridors for NTC

Morphological Operations

Morphological operators define operations that can be applied to either a binary or continuous image (Mazzocchi 1991). For each morphological operation, a structuring element must be defined. A structuring element usually consists of a few pixels in a specified configuration. This structuring element is applied to each pixel of the base image to yield a new image that contains the desired information. This process is demonstrated with a sample binary image like the one that would be input to the thinning algorithm. Figure 9 shows the sample binary image A and a 2 by 2 structuring element B. The result of applying the morphological dilation operator to image A is shown in Figure 10. The dilation operation resembles the mathematical addition operation. In the dilation operation, if a pixel in the image is set to one, then all surrounding elements corresponding to where the structuring element is one are set to one. The result of applying the morphological erosion operator to image A is shown in Figure 11. The erosion operation resembles the mathematical subtraction operation. In the erosion operation, if a pixel in the image is set to one and all surrounding elements corresponding to where the structuring element is one are also set to one, then the pixel remains one otherwise the pixel is set to zero.

The morphological close operation is formed by application of dilation and then erosion operators to an image using the same structuring element. Figure 12 shows the result of performing a morphological close on image A. The use of a morphological close eliminates specific image detail smaller than the structuring element. The morphological close operation is run on the binary speed matrix when no resolution change is performed. Figure 13 shows the effect of performing a morphological close on the binary speed matrix prior to thinning. The structuring element used to create Figure 13 was a 2 by 2 matrix as shown in Figure 9. As shown in Figure 13, fewer edges are created when the morphological close is run prior to thinning. But, there is a risk that the morphological close operation will, at times, allow the thinned areas to cross small areas of speed below the desired threshold.

Data Resolution Modification

The resolution of the input speed matrix is modified if either of two criteria is met. The first criterion would be met if the input matrix had one or both dimensions greater than the preset parameters in the software. In this instance the software would compute a square dimension (i.e., 2 by 2, 3 by 3, etc.) for resolution modification that would produce a matrix smaller than the maximum dimension. The second criterion involves the use of a user-input parameter. This parameter allows the user to designate the desired minimum unit for which MCs are computed. The resolution would be set to the closest square dimension to the width of the desired MC, divided by four. The value assigned to the resultant larger cells is based on the percentage of zero cells found within the corresponding input cells. Currently, if 50 percent or greater of the corresponding input cells are zero, then the cell will be set to zero otherwise it will be set to one. This larger cell resolution would result in fewer MCs for units lower than the desired

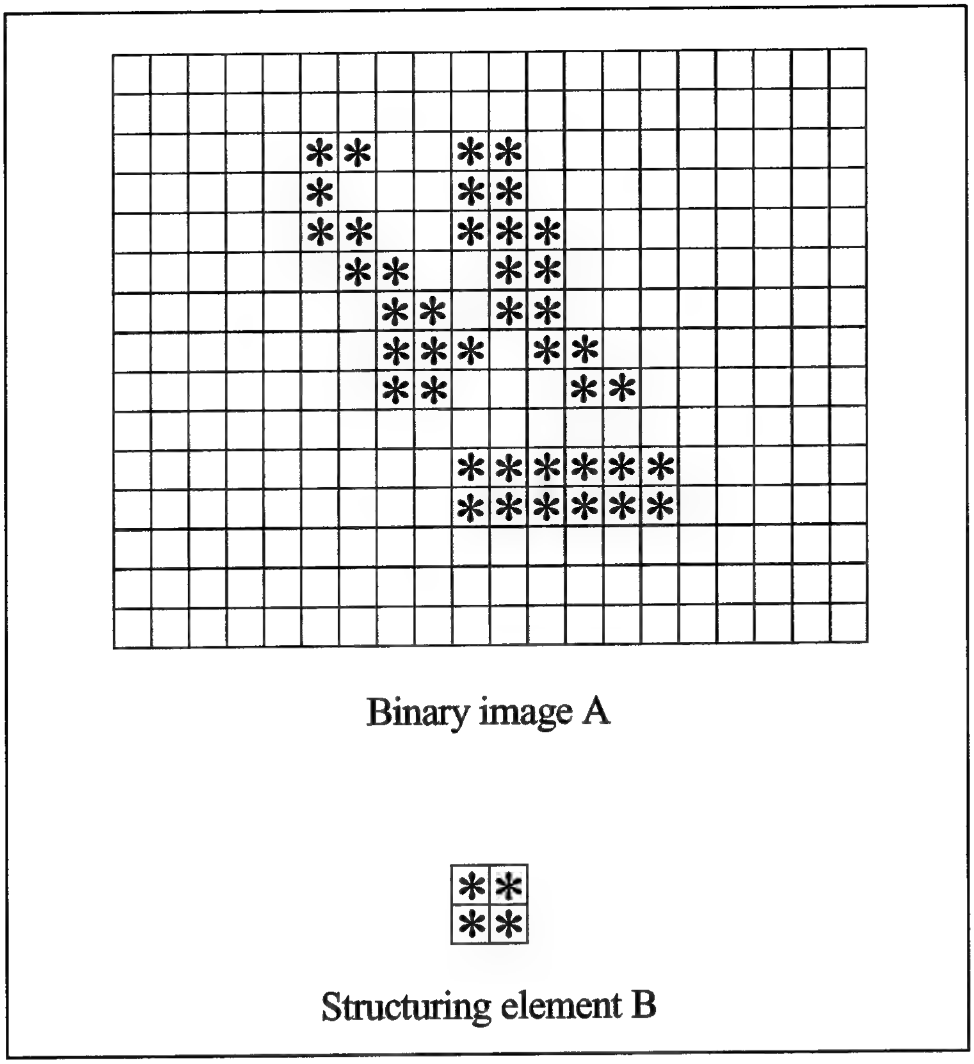

Figure 9. Sample binary image and structuring element

minimum unit. Example MC networks for battalions, brigades, and divisions for the NTC are shown in Figures 14, 15, and 16, respectively. For these examples, speeds in urban areas and speeds less than or equal to 1 kph were set to zero.

Generation of Voronoi Diagrams by Means of Patch Expansion

The mobility corridor generation software developed at Los Alamos makes use of a Voronoi diagram. The Voronoi diagram is also known as a Thiessen diagram and is referred to as such in the documentation of the software (Powell et al. 1988). A Voronoi diagram is essentially a partition of a plane into polygonal regions, each of which is associated with a given point. The region associated with a point is the locus of points closer to that point than to any other given point. Vector-based algorithms, such as the one created at Los Alamos, are complex and

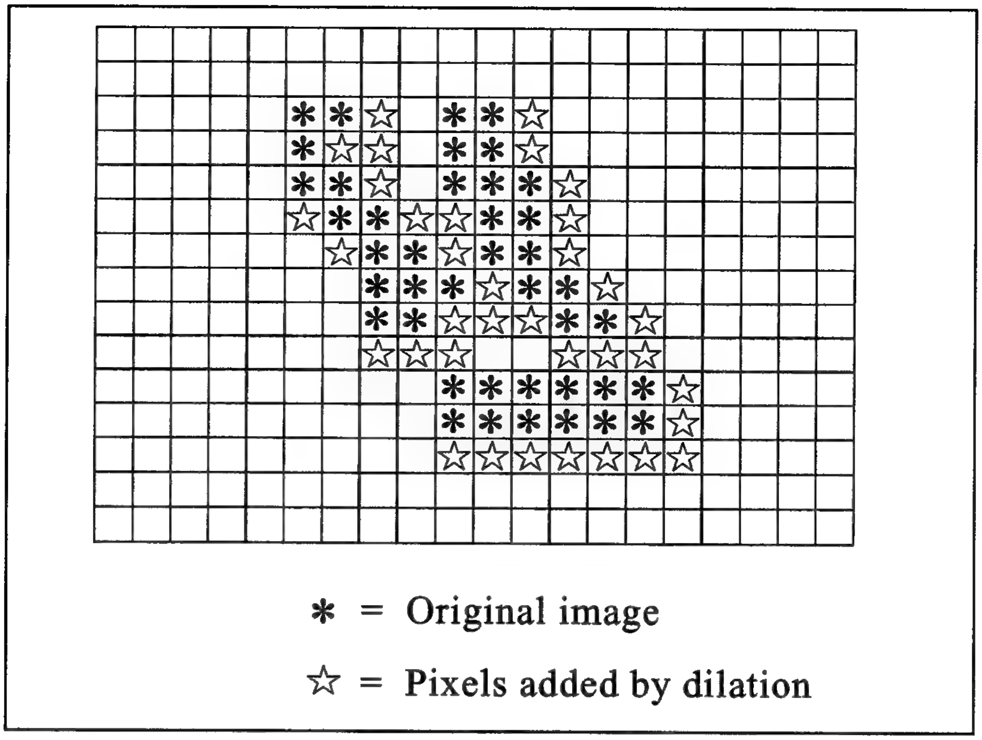

Figure 10. Result of dilation of A by B

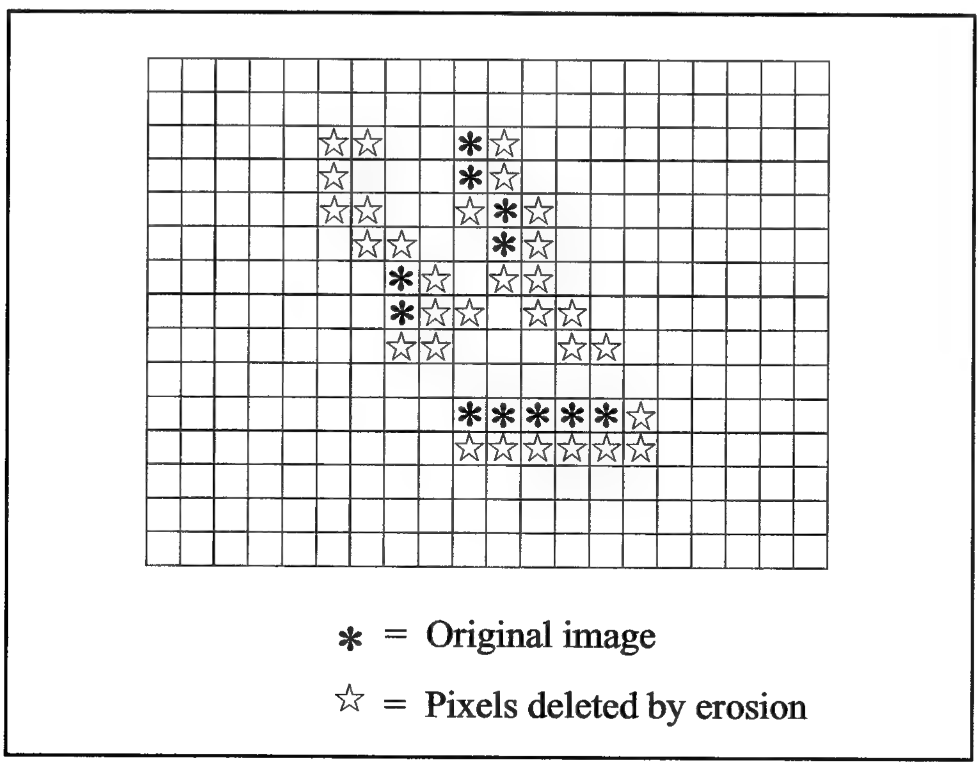

Figure 11. Result of erosion of A by B.

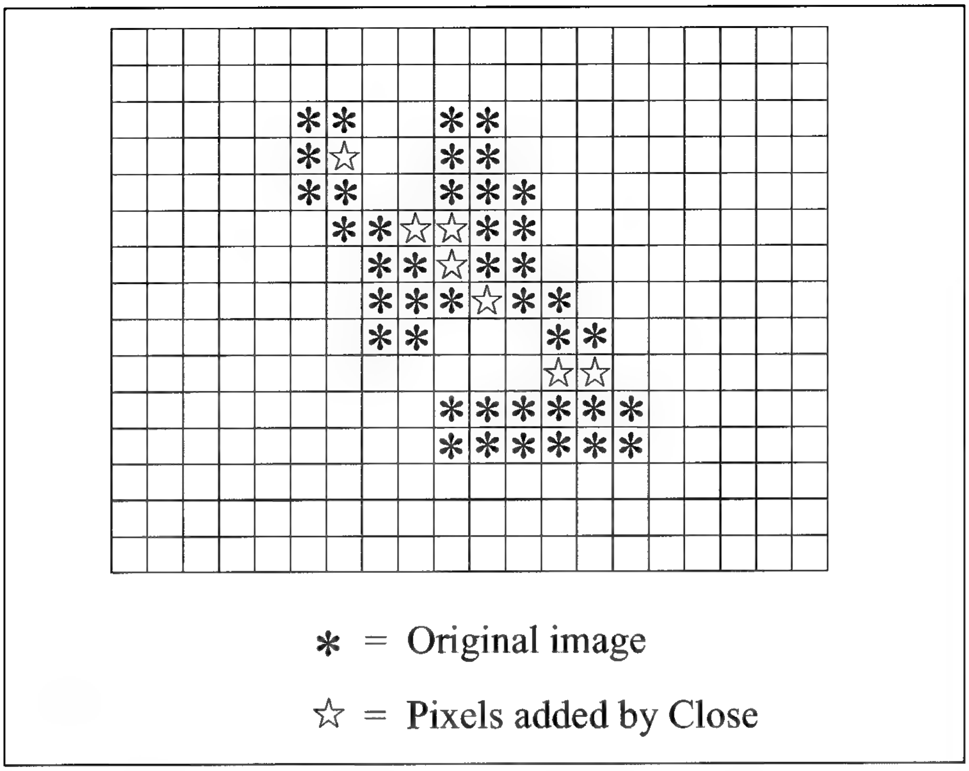

Figure 12. Result of closing A by B

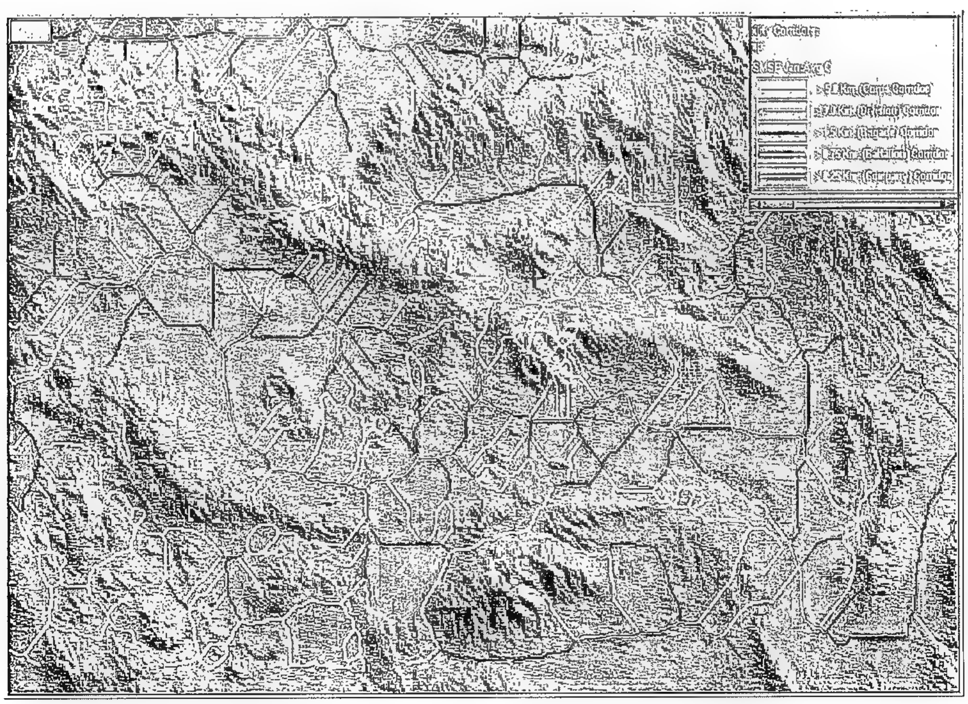

Figure 13. Mobility corridors for NTC after morphological close

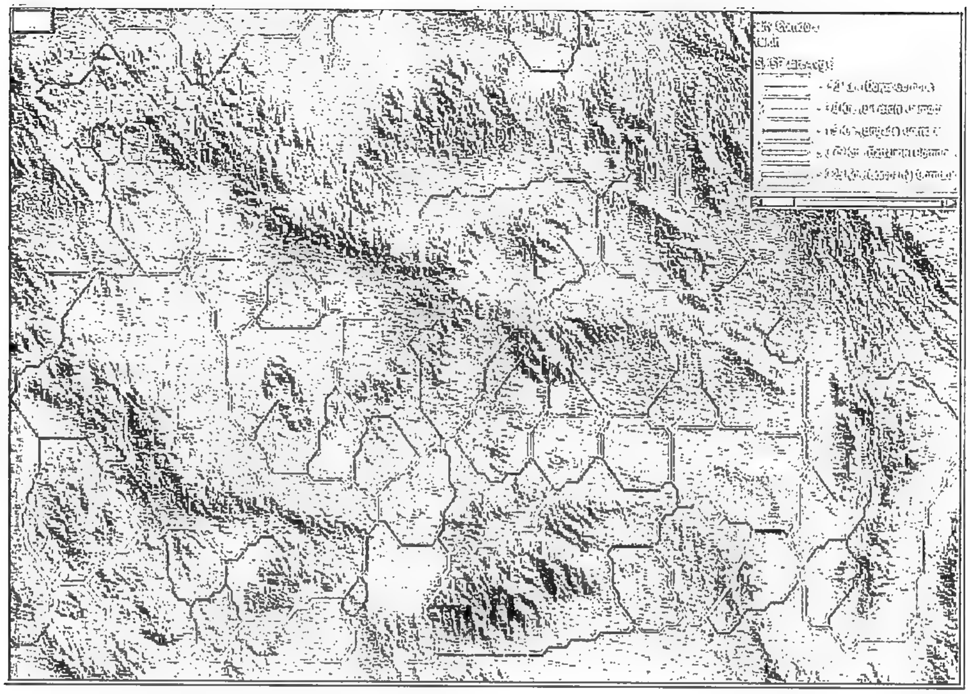

Figure 14. Mobility corridors when battalion and above corridors are desired

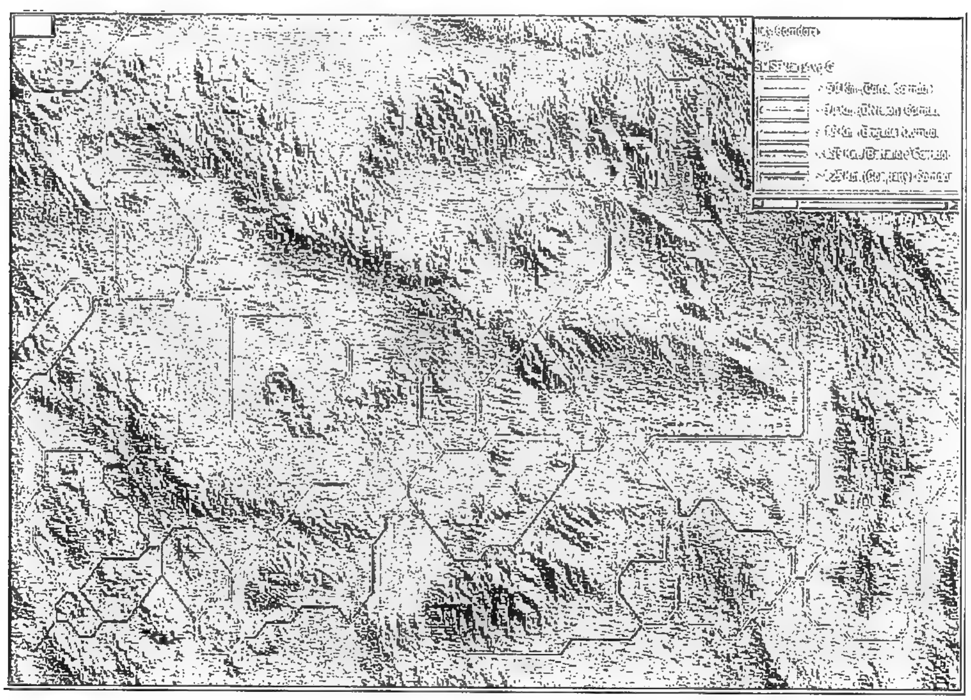

Figure 15. Mobility corridors when brigade and above corridors are desired

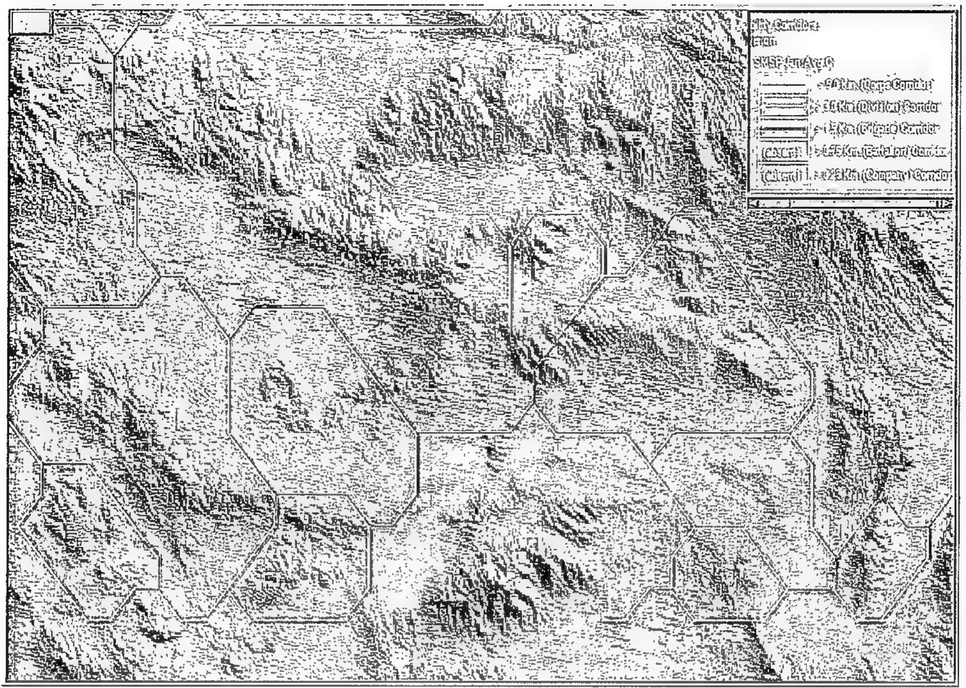

Figure 16. Mobility corridors when division and above corridors are desired

computationally intensive for line sets and area sets. One raster-based method that has been developed uses dynamic distance transformation achieved by the dilation operator in mathematical morphology (Li et al. 1999). The patch expansion algorithm presented in this report can also be a raster-based generator of Voronoi diagrams. The major disadvantage of patch expansion in generating mobility corridors becomes an advantage in Voronoi diagram generation. The dead-end edges falling within a patch produced by modified classical thinning are not desired in a Voronoi diagram. The first change necessary to the patch expansion algorithm is that the one-pixel-wide buffer around the edge of the map is set to zero instead of one. The patches could also be numbered sequentially starting at one, instead of at two. The output produced by this modification to the patch expansion algorithm on the test case is shown in Figure 17. The edges clearly create a region for each input patch. Figure 18 shows a comparison of the output produced by patch expansion and a Voronoi diagram produced by a vector-based method when only points are used as input. Figure 18 shows the output from patch expansion with only the end points (nodes) of each edge plotted.

The results shown in Figure 18 are promising. A further enhancement would be to disallow diagonal expansion on every pass through the map in order to get each pair of expanding patches to meet at a point equidistant from each other. On each pass the pass counter is compared to a variable S which was initialized to the square root of 2. If the pass counter were greater, then no diagonal expansions would occur on this pass. Otherwise diagonal expansions would be allowed and the variable S would be incremented further by the square root of 2. Figure 19 shows a comparison of the output produced by this modified code and a Voronoi diagram produced by a vector-based method. The figure also

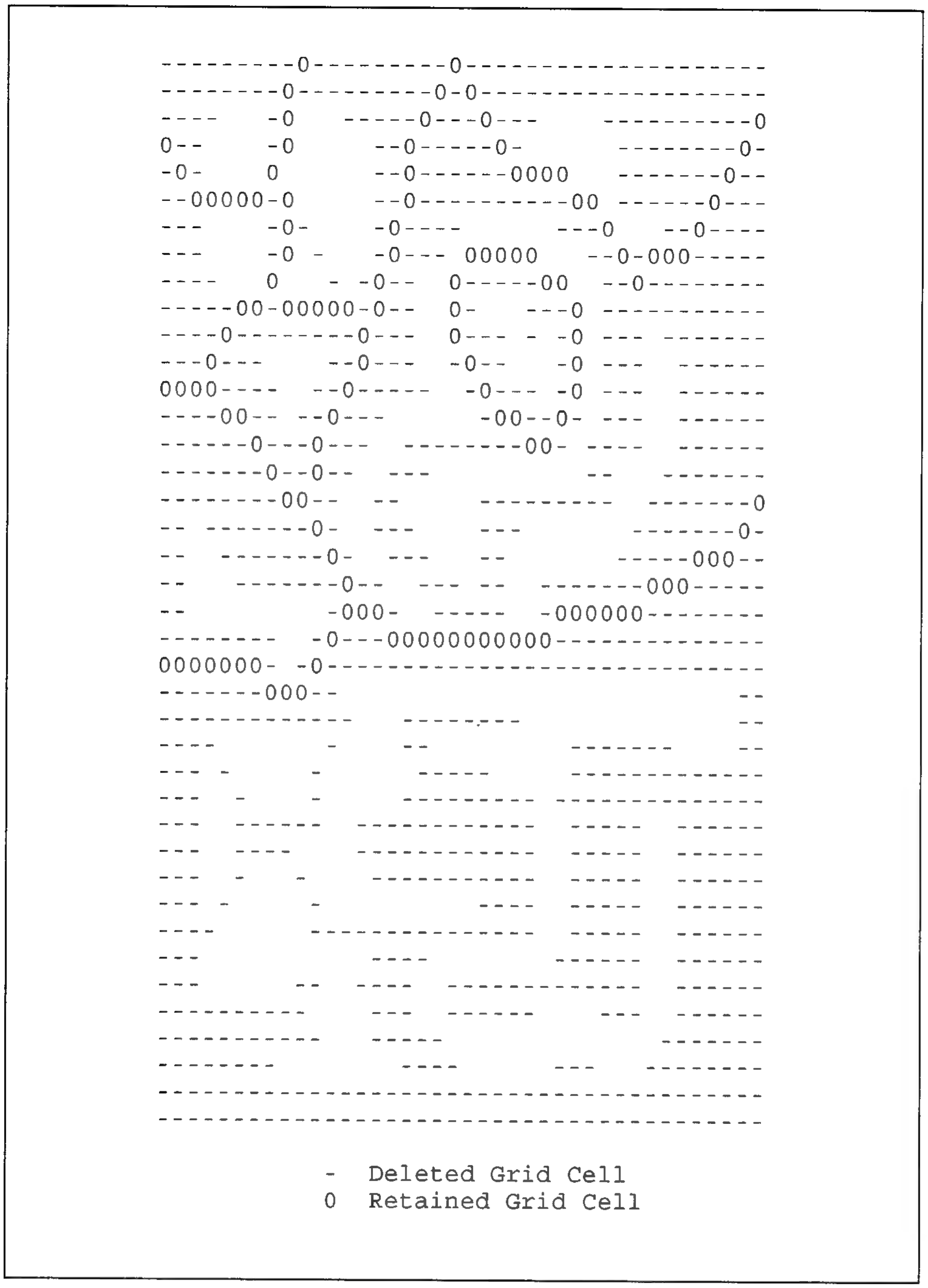

Figure 17. Voronoi diagram for test case

shows the output from patch expansion with only the end points (nodes) of each edge plotted. This code modification produced results closer to the vector-based method and appears to be a good approximation indicating an algorithm which appears to be a viable method of producing Voronoi diagrams for large data sets that include points, lines, and areas.

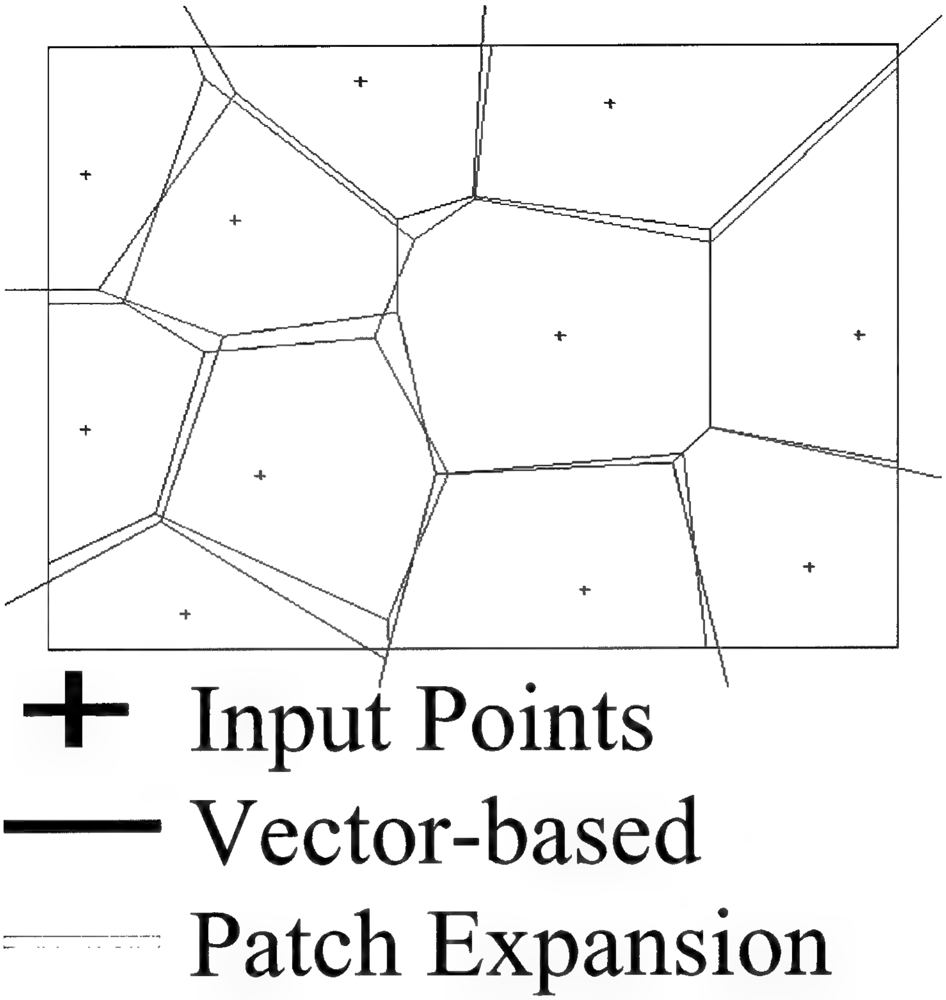

Figure 18. Voronoi diagrams using a vector-based method and patch expansion

Additional Edge Attributes

The Eagle database requires some edge attributes that were not computed in the TEM MC TDA. These attributes are a minimum width, average width, length, mobility ratio, concealment ratio, and cover ratio for each MC edge. The minimum width is computed by converting the minimum number of thinning passes required for any cell in the edge to meters. The average width is computed by converting the average number of thinning passes required for all cells in the edge to meters. The length in meters of the edge is computed by summing the lengths of the individual vectors comprising the edge. The mobility ratio is the number of cells containing GO speed values divided by the total number of cells which fall beneath the area encompassed by each MC. The concealment ratio is the number of cells that provide concealment divided by the total number of cells that fall beneath the area encompassed by each MC.

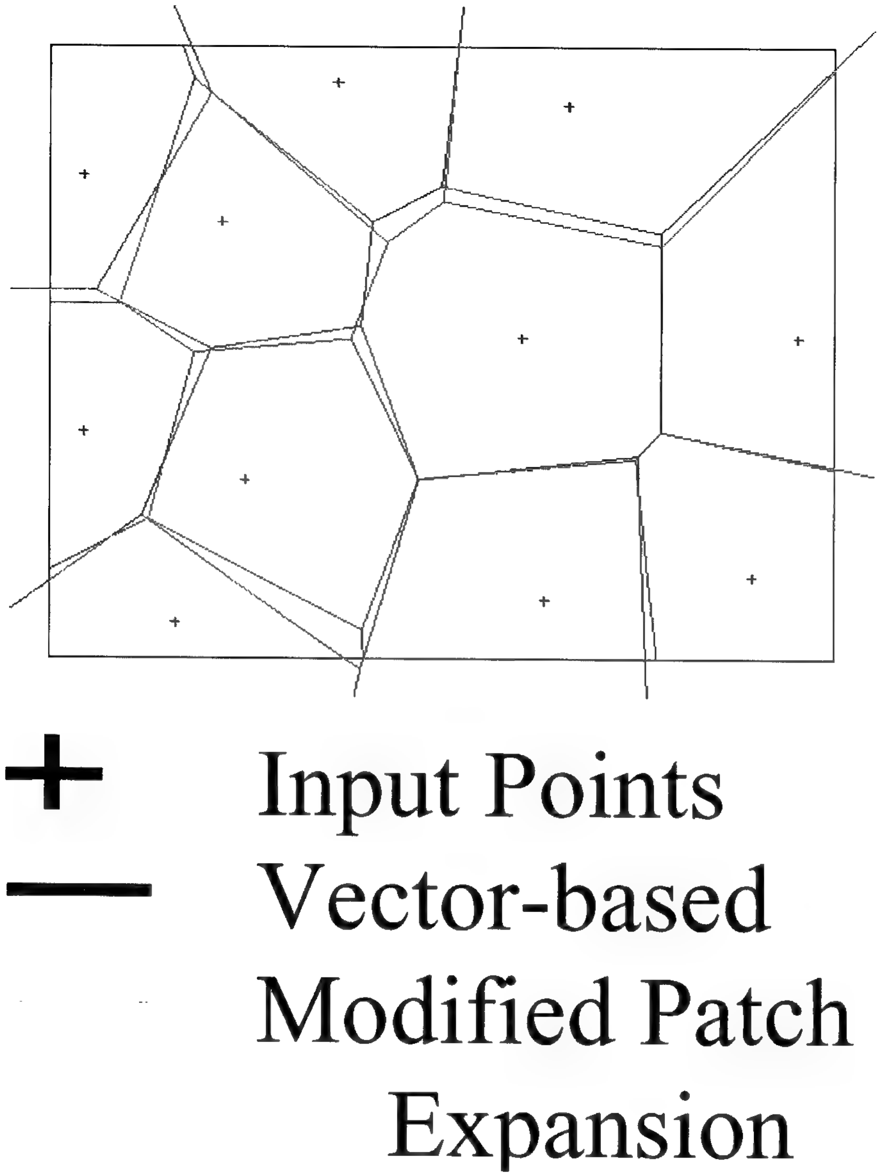

Figure 19. Voronoi diagrams using a vector-based method and modified patch expansion

A cell is deemed to be concealed if either one of two criteria are met. The first criterion is met if any of the eight surrounding cells has an elevation difference of 6 m or more when compared to the cell in question. The second criterion is met if the cell has vegetation height greater than 3 m, spacing greater than forty decimeters and canopy closure of greater than 50 percent. The cover ratio is assigned the same value as the concealment ratio. TEM saves these new attributes in the TEM Geographic Information System (GIS), allowing viewing of the attributes. TEM has the capability to output ASCII edge and node files in Eagle format. The file formats and examples are shown in Figures 20 and 21.

MC-EDGE.# {edge name}
((X,Y) (X,Y) … (X,Y)) {coordinates of edge}
Minimum Width {meters}
Average Width {meters}
Length {meters}
Mobility {ratio of GO cells to total cells covered by edge}
Concealment {ratio of cells providing concealment to total cells covered by edge}
Cover {ratio of cells providing cover to total cells covered by edge}
NIL {not used}

```
MC-EDGE.0
((530980.000000 5626660.000000) (530980.000000
5626760.000000) (531280.000000 5627060.000000)
(531280.000000 5627160.000000) (531380.000000
5627260.000000) (531380.000000 5627360.000000)
(531080.000000 5627660.000000) (530980.000000
5627760.000000) (530680.000000 5627760.000000)
(530580.000000 5627660.000000) (530380.000000
5627660.000000) (530280.000000 5627660.000000)
(530080.000000 5627460.000000) (530080.000000
5627360.000000))
200.000000
445.000000
2551.000000
0.923700
0.842100
0.842100
NIL
```

Figure 20. Format and example for edge file.

```
MC-NODE.# {node name}
(X,Y) {coordinate of node}
(MC-NODE.# MC-NODE.# MC-NODE.#) {ends of adjoining edges}
(MC-EDGE.# MC-EDGE.# MC-EDGE.#) {list of adjoining edges}


MC-NODE.0
(530980.000000 5626660.000000)
(MC-NODE.1 MC-NODE.7 MC-NODE.27)
(MC-EDGE.0 MC-EDGE.22 MC-EDGE.408)
MC-NODE.1
(530080.000000 5627360.000000)
(MC-NODE.0 MC-NODE.6 MC-NODE.7)
(MC-EDGE.0 MC-EDGE.4 MC-EDGE.5)
MC-NODE.2
(543880.000000 5627560.000000)
(MC-NODE.3 MC-NODE.4 MC-NODE.5)
(MC-EDGE.1 MC-EDGE.2 MC-EDGE.3)
MC-NODE.3
(543080.000000 5627260.000000)
(MC-NODE.2 MC-NODE.8 MC-NODE.5)
(MC-EDGE.1 MC-EDGE.6 MC-EDGE.8)
MC-NODE.4
(544480.000000 5626760.000000)
(MC-NODE.2 MC-NODE.12 MC-NODE.40)
(MC-EDGE.2 MC-EDGE.26 MC-EDGE.434)
```

Figure 21. Format and example for node file.

3 Aggregated Terrain Tile and Background Map Generation

Terrain Tile Aggregation

The aggregated terrain file is composed of a header record followed by one record for each terrain aggregate (TERAG). The header record contains the following:

a. Universal Transverse Mercator (UTM) coordinates of the lower left origin

b. X and Y size of terrain box in meters

c. X and Y size of individual TERAG in meters

d. Military Grid Reference of lower left origin

e. Spheroid upon which the data is based

Each aggregated terrain tile record consists of the following:

a. TERAG index

b. Coordinates of lower left corner of TERAG usually in UTM coordinates

c. Relief classification

d. Mobility ratios (GO, SLOWGO, and NOGO)

e. Vegetation ratios (urban, dense, medium and light vegetation)

f. Mean in-view path length

g. Line of Sight (LOS) alphas for three observer heights (2, 4 and 6 m)

The geometry of Eagle calls for a rectangular coordinate system with an underlying rectangular map. The map coordinates are based on the UTM grid system. If the desired area-of-interest (AOI) is included in more than one six-degree UTM grid zone, the user must select from among the zones covered by the AOI. It would be

advisable to select a grid zone that lies near the center of the AOI for extension, since this would minimize warping of the data at the extremities of the AOI. The TERAGs are overlaid on this grid and are anchored at the upper left corner of the map. The anchoring is important if an extent of the full map is not a multiple of the corresponding X or Y size of the individual TERAG. In this case the bottom and/or right rows of TERAGs are not the full size, but are whatever size is needed to fill in the map. Assignment of the TERAG index is best viewed as starting with the lower left corner. The lower left TERAG is labeled 0-0, its northern neighbor is labeled 0-1, and its eastern neighbor is labeled 1-0, etc.

Relief is classified as either mountainous, rolling or flat. Raw elevation data for each point on a grid within the TERAG are used for this classification. The maximum difference between two contiguous points in the TERAG is found. If the maximum difference is greater than 100 then the relief is mountainous; if the maximum difference is greater than 30 then the relief is rolling; otherwise the relief is classified as flat.

The mobility ratios are simply the ratio of the number of points in a category (GO, SLOWGO and NOGO) to the total number of points in the TERAG. If a point has a speed less than 8 kph, it is classified as NOGO; if its speed falls between 8 kph and 25 kph it is classified as SLOWGO; otherwise it is classified as GO.

The vegetation ratios are simply the ratio of the number of points in a category (urban, dense, moderate and light vegetation) to the total number of points in the TERAG. If a point falls in an urban area, it is added to the urban counter. Otherwise, if the point has a tree spacing density less than 133 decimeters, it is classified as densely vegetated; if its tree spacing density falls between 133 decimeters and 266 decimeters, it is classified as moderately vegetated; otherwise it is classified as lightly vegetated.

A YES/NO determination of line-of-sight (Powell 1988) is made for each of 64 evenly spaced points on a grid within the TERAG. This determination of line-of-sight is made for each of three observer heights (2, 4 and 6 m) at each of fifteen ranges (300, 600, 900 ... and 4,500 m) on each of 36 points on a circle around the view point. Thus, yielding a sample of some 103680 determinations for each full sized TERAG. A target height of 2 m is used and the geometric determination of line-of-sight for each point is made, based on terrain elevation. Vegetation heights are added to intervening elevations. These samples provide estimates of probability of line-of-sight (*plos*) for the TERAG for each of the fifteen ranges. These are used to develop a least squares estimate of *alpha* in the equation:

$$plos = e^{-(alpha \times range)}$$

In this equation *range* is given in kilometers. This equation is used to determine *plos* within Eagle. The least squares estimate for *alpha* is:

$$alpha \approx \frac{-\sum_{i=1}^{15}\left(range_i \times \ln\left(plos_i\right)\right)}{\sum_{i=1}^{15} range_i^2}$$

For this computation *range* is also in kilometers. The computations are made independently for each observer height. Given the previous computations, the value of *alpha* for each of the three viewer heights is used in addition to a mean in view path length. This mean in view path length is computed using the *alpha* from the 2-m observer height as follows:

$$mean = 405.81 - 324.27 \times \ln(alpha)$$

This mean in view path length is in meters.

Background Map Creation

The Eagle background map is merely an X-Window dump of a vehicle speed map with roads and drainage also displayed. The X-Window dump utility developed for Eagle allows the user to enter the maximum desired size of the output matrix in both the X and Y dimensions. The user may then select the cross-country movement evaluation on which the output is to be based. The user also selects the maximum road type to be displayed from among super highway, primary, secondary, and trails. For example, if the user selected secondary, then all roads classified as secondary or lower (i.e., super highways, primary, and secondary roads) would be displayed in the resulting X-Window dump file. The user also assigns a minimum drainage width falling between one and 1,000 m. Only those drainage features with widths greater than or equal to the minimum will appear in the generated output. The road and drainage limiting criteria allow the user to minimize the amount of clutter in the resulting background map.

4 Conclusions and Recommendations

Conclusions

Based on the results of this investigation, the following conclusions are drawn:

a. The modified classical thinning algorithm was the best suited of the three algorithms tested for the application of automatic MC generation.

b. The software to generate an Eagle terrain data set now resides on TEM, which is an element of the Army Common Hardware/Software Program and is already operational at CAA.

c. This software's parameters are modifiable by input to forms, which is far superior to the previous software which often required code modifications whenever a new area was selected for use in Eagle.

d. Although no direct comparisons to the previous software are possible, it is evident from testing that the TEM implementation is much faster.

e. The patch expansion algorithm produced good results when applied to the generation of Voronoi diagrams.

Recommendations

Based on the information presented in this study, it is recommended to:

a. Accept the software presented in this report as a preprocessor for the Eagle model.

b. Investigate the potential of this Eagle preprocessor as a starting point in the development of a scalable terrain generation system to support potential future Eagle/theater simulation and support a possibly scalable JWARS model in the future.

References

Li, C., Chen, J., and Li, Z. (1999). "A raster-based method for computing voronoi diagrams of spatial objects using dynamic distance transformation," *International Journal of Geographical Information Science* 13(3), 209-225.

Mazzocchi, F. (1991). "The air defense deployment support system AI and ADA," AI Exchange, Vol. 5, No. 4, U.S. Military Academy, West Point, NY.

McKinley, G. B., Butler, C. D., Horner, D. A., Robinson, J. H., and Dickson, W. C. (1993). "Enhanced automated selection and evaluation of mobility corridors and isochrone computation," Technical Report GL-93-12, U.S. Army Engineer Waterways Experiment Station, Vicksburg, MS.

Pavlidis, T. (1982). *Algorithms for graphics and image processing*. Computer Science Press, Inc., Rockville, MD.

Powell, D. R. (1988). CALC-ALP, A line-of-sight probability routine (computer program), UNIX platform, Los Alamos National Laboratory, Los Alamos, NM.

Powell, D. R., Wright, J. C., Slentz, G., and Knudsen, P. (1988). "Representations to support reasoning on terrain," Unpublished Report LA-UR-88-3754, Los Alamos National Laboratory, Los Alamos, NM.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
September 2000

3. REPORT TYPE AND DATES COVERED
Final report

4. TITLE AND SUBTITLE
Eagle Terrain Preprocessor

5. FUNDING NUMBERS
AMIP-97-WARF-0

6. AUTHOR(S)
George B. McKinley, Terril C. Falls, David C. Stuart

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Engineer Research and Development Center, Geotechnical Laboratory,
3909 Halls Ferry Road, Vicksburg, MS 39180-6199;
Applied Research Associates, Vicksburg, MS 39180

8. PERFORMING ORGANIZATION REPORT NUMBER
ERDC/GL TR-00-10

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Corps of Engineers
Washington, DC 20314-1000

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The purpose of this report is to describe the automated user-friendly programs produced by the U.S. Army Engineer Research and Development Center, Vicksburg, MS, which create the terrain-related inputs required by the Eagle model. The project was divided into three tasks. The first task involved modification of the mobility corridors (MC) network creation software in Terrain Evaluation Module (TEM) to compute the additional attributes required by the Eagle model and also to output the network in the format required by Eagle. Also, three methodologies are examined to discover the approach that will create the most complete MC network. The second task was to produce software to create the Eagle terrain tiles and output them in the proper format. Task three required the creation of software to generate a graphics image in X-Windows dump format for use as a background image by Eagle. These programs are described in terms of their input, methodology, and output.

Based on the results of this investigation, the following conclusions are drawn:

a. The modified classical thinning algorithm was the best suited of the three algorithms tested for the application of automatic MD generation.

(Continued)

14. SUBJECT TERMS

Line of sight	Thinning algorithms
Mobility corridors	Voronoi diagrams

15. NUMBER OF PAGES
34

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED		

13. (Concluded).

b. The software to generate an Eagle terrain data set now resides on TEM, which is an element of the Army Common Hardware/Software Program and is already operational at the Center for Army Analysis.

c. The parameters for this software are modifiable by input to forms, which is far superior to the previous software that often required code modifications whenever a new area was selected for use in Eagle.

d. Although no direct comparisons to the previous software are possible, it is evident from testing that the TEM implementation is much faster.

e. The patch expansion algorithm produced good results when applied to the generation of Voronoi diagrams.

Based on the information presented in this study, it is recommended to:

a. Accept the software presented in this report as a preprocessor for the Eagle model.

b. Investigate the potential of this Eagle preprocessor as a starting point in the development of a scalable terrain generation system to support potential future Eagle/theater simulation and support a possibly scalable JWARS model in the future.